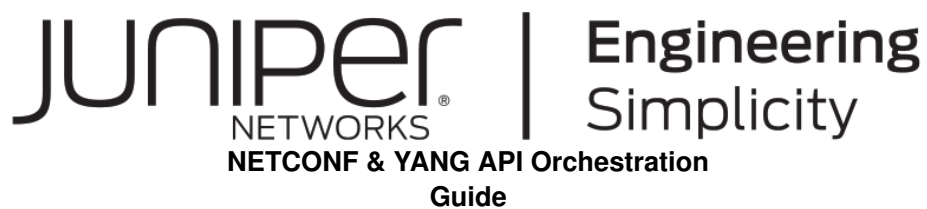
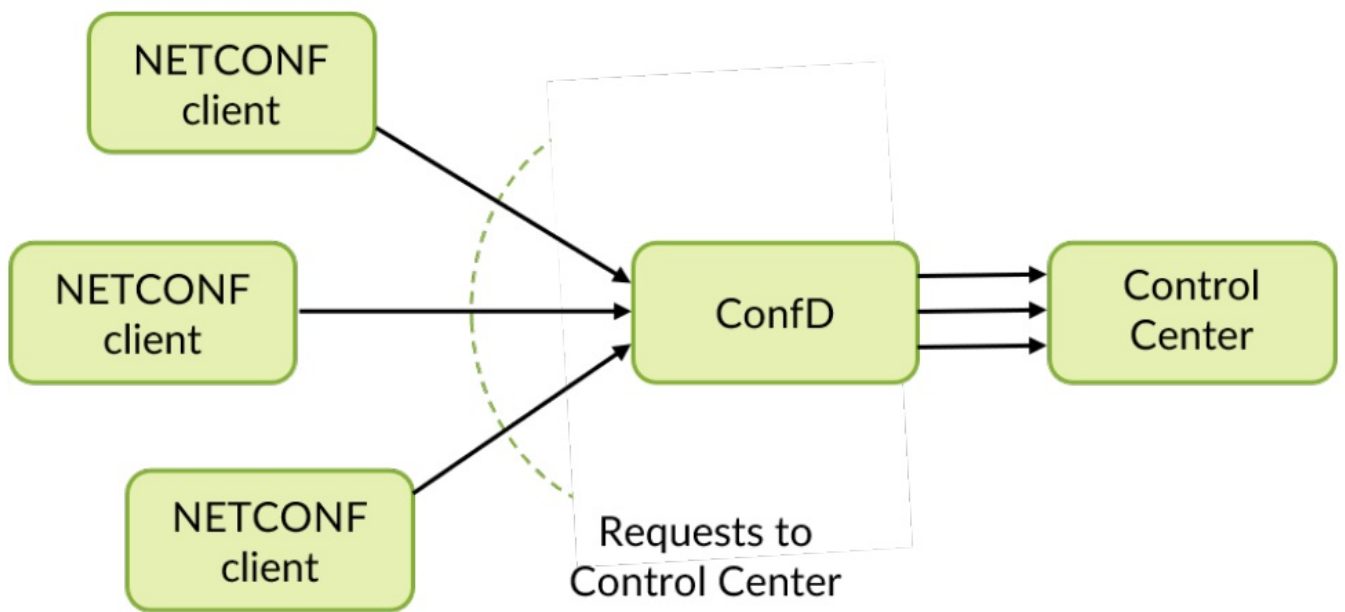




Juniper NETWORKS NETCONF & YANG API Software User Guide

[Home](#) » [JUNIPER NETWORKS](#) » Juniper NETWORKS NETCONF & YANG API Software User Guide 





```
#
# The script is not meant to run on the NCC server!
# Run it on a separate client computer.
#

from ncclient import manager
from ncclient.xml_ import to_ele

# NETCONF server
host = '<host IP>'          # Replace with server IP of Control Center
port = 830
user = '<username>'        # Replace with username defined in "netconf user create"
command
password = '<password>'    # Replace with password defined in "netconf user create"
command

# Product account
netrounds_account = '<account>' # Replace with account name

with manager.connect(host=host, port=port, username=user, password=password,
                    hostkey_verify=False) as m:

    # Update config database
    xml = """<sync-from-ncc xmlns="http://ncc.netrounds.com"/>"""

    # Convert to ElementTree Element
    elem = to_ele(xml)
    print m.dispatch(elem)
```

Account & settings



Permissions

Invite colleagues and change user permissions.



admin@example.org

Director admin

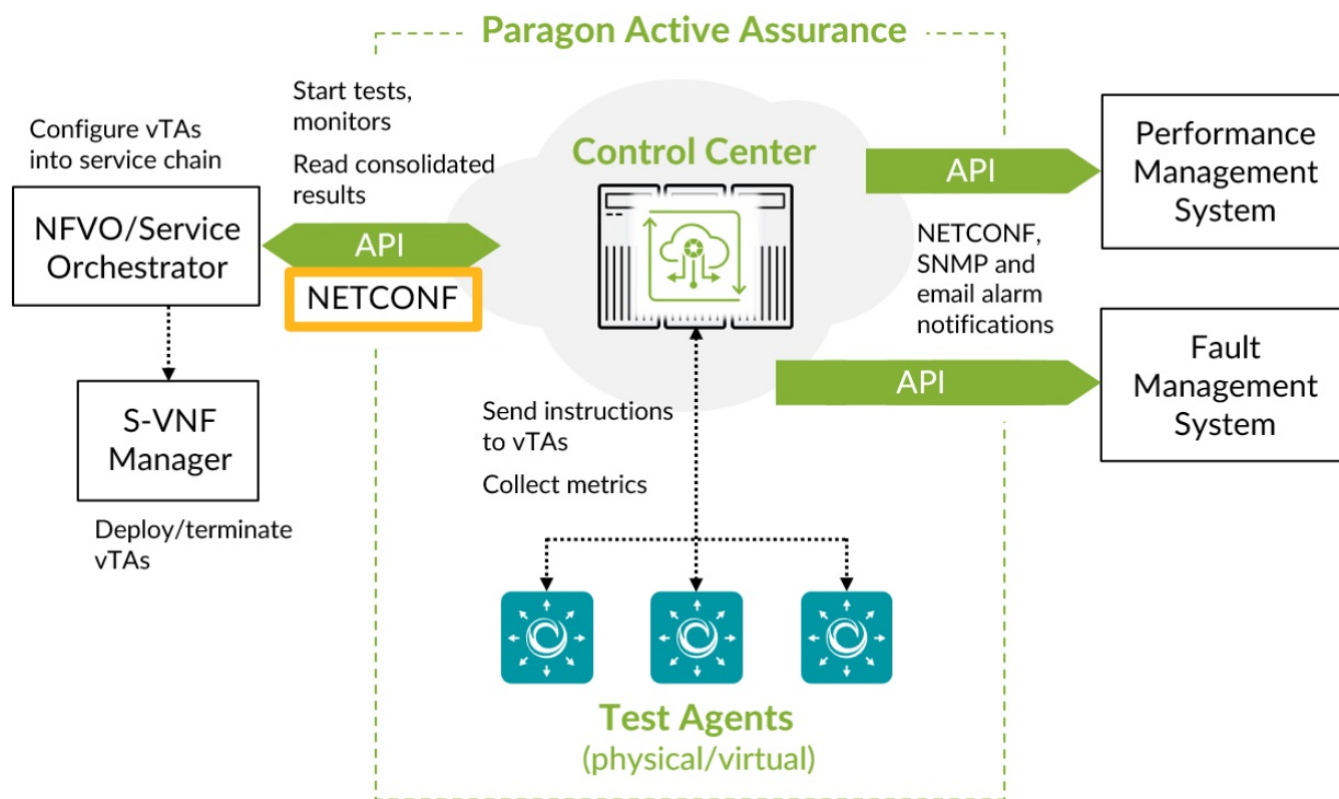
admin



confd@netrounds.com

Administrator ▼

Invite



Feature	Subfeature	YANG construct
Manage TWAMP reflector inventory	Create/modify/delete TWAMP reflectors	Config

YANG path: /accounts/account/y1731-meps

Feature	Subfeature	YANG construct
Manage Y.1731 MEP inventory	Create/modify/delete MEPs	Config

YANG path: /accounts/account/iptv-channels

Feature	Subfeature	YANG construct
Manage IPTV channel inventory	Create/modify/delete IPTV channels	Config

YANG path: /accounts/account/ping-hosts

Feature	Subfeature	YANG construct
Manage Ping host inventory	Create/modify/delete Ping hosts	Config

YANG path: /accounts/account/sip-accounts

Feature	Subfeature	YANG construct
Manage SIP account inventory	Create/modify/delete SIP accounts	Config

Feature	Subfeature	YANG construct
Configure alarm email lists	Create/modify/delete alarm email lists	Config

YANG path: /accounts/account/snmp-managers

Feature	Subfeature	YANG construct
Configure SNMP managers	Create/modify/delete SNMP managers	Config

Resource: Global Statistics

YANG path: /statistics

Feature	Subfeature	YANG construct
Retrieve statistics for entire Control Center	--	Config

Resource: Account Statistics

YANG path: /accounts/account/statistics

Feature	Subfeature	YANG construct
Retrieve statistics for specific account	--	Config

Feature	Subfeature	YANG construct
Configure alarm email lists	Create/modify/delete alarm email lists	Config

YANG path: /accounts/account/snmp-managers

Feature	Subfeature	YANG construct
Configure SNMP managers	Create/modify/delete SNMP managers	Config

Resource: Global Statistics

YANG path: /statistics

Feature	Subfeature	YANG construct
Retrieve statistics for entire Control Center	--	Config

Resource: Account Statistics

YANG path: /accounts/account/statistics

Feature	Subfeature	YANG construct
Retrieve statistics for specific account	--	Config



```

+--rw test-agents
| +--rw test-agent* [name]
|   +--rw name          string
|   +--rw description?  string
|   +--rw tags
|   | +--rw tag* [name]
|   |   +--rw name      -> ../../../../tags/tag/name
|   +--ro state
|   | +--ro status?      test-agent-status-t
|   | +--ro version?     string
|   | +--ro uptime?      int64
|   | +--ro memory-usage? percent-t
|   | +--ro load-average
|   |   +--ro last-1-minute?  decimal64
|   |   +--ro last-5-minutes? decimal64
|   |   +--ro last-15-minutes? decimal64
|   +--rw ntp
|   | +--rw interface?   string
|   | +--rw servers* [address]
|   | | +--rw address    union
|   | +--rw enable-ipv6? boolean
|   +--rw management
|   | +--rw interface?   string
|   | +--rw use-public-address? boolean
|   +--rw interfaces
|   | +--rw interface* [name]
|   |   +--rw name          string
|   |   +--rw description?  string
|   |   +--rw interface-type? interface-type-t
|   |   +--rw mac?          yang:mac-address
|   |   +--rw mtu?          uint16
|   |   +--rw speed?        speed-t
|   |   +--rw bridge

```

```

| | | +---rw members
| | |   +---rw member* [interface]
| | |     +---rw interface -> ../../../../interface/name
| | | +---rw vlan
| | |   +---rw id          vlan-id-t
| | |   +---rw parent      -> ../../../../interface/name
| | | +---rw wifi
| | |   +---rw ssid?       string
| | |   +---rw bssid?      yang:mac-address
| | |   +---rw country?    string
| | |   +---rw ht          boolean
| | |   +---rw ht40        boolean
| | |   +---rw vht         boolean
| | |   +---rw sgi         boolean
| | |   +---rw ldpc        boolean
| | |   +---rw ht-mcs?     string
| | |   +---rw vht-max-mcs uint8
| | |   +---rw vht-max-mimo uint8
| | |   +---rw freq-2-4    boolean
| | |   +---rw freq-5      boolean
| | |   +---rw auth
| | |     +---rw (auth-type)?
| | |       +---:(personal)
| | |         | +---rw personal!
| | |         |   +---rw cipher      cipher-auth-t
| | |         |   +---rw password?  string
| | |         +---:(eap-tls)
| | |           | +---rw eap-tls!
| | |           |   +---rw cipher      cipher-auth-t
| | |           |   +---rw ca-cert?    string
| | |           |   +---rw client-cert? string
| | |           |   +---rw identity?   string
| | |           |   +---rw private-key? string
| | |           +---:(eap-ttls)
| | |             | +---rw eap-ttls!
| | |             |   +---rw cipher      cipher-auth-t
| | |             |   +---rw ca-cert?    string
| | |             |   +---rw identity?   string
| | |             |   +---rw anonymous-identity? string
| | |             |   +---rw password?  string
| | |             +---:(peap)
| | |               +---rw peap!
| | |               +---rw cipher      cipher-auth-t

```

```

| | | +--rw ca-cert?      string
| | | +--rw identity?     string
| | | +--rw anonymous-identity? string
| | | +--rw password?     string
| | +--rw mobile
| | | +--rw apn?          string
| | | +--rw pdp_type      string
| | | +--rw rat_band      mobile-rat-band-t
| | | +--rw rat_mode      mobile-rat-mode-t
| | +--rw interface-config
| | | +--rw ipv4-address
| | | | +--rw (address-type)?
| | | | +--:(static)
| | | | | +--rw static!
| | | | | | +--rw address      tailf:ipv4-address-and-prefix-length
| | | | | | +--rw routes* [network]
| | | | | | | +--rw network    tailf:ipv4-address-and-prefix-length
| | | | | | | +--rw gateway    inet:ipv4-address
| | | | | | +--rw dns-servers* inet:ipv4-address
| | | | | +--:(dhcp)
| | | | | +--rw dhcp!
| | | | | | +--rw vendor-id?   string
| | | +--rw ipv6-address
| | | | +--rw (address-type)?
| | | | +--:(static)
| | | | | +--rw static!
| | | | | | +--rw address      tailf:ipv6-address-and-prefix-length
| | | | | | +--rw routes* [network]
| | | | | | | +--rw network    tailf:ipv6-address-and-prefix-length
| | | | | | | +--rw gateway    inet:ipv6-address
| | | | | | +--rw dns-servers* inet:ipv6-address
| | | | | +--:(dhcp)
| | | | | +--rw dhcp!
| | | | | | +--rw vendor-id?   string
| | | | +--:(slaac)
| | | | +--rw slaac!
| | | | | +--rw dns-servers*   inet:ipv6-address
| +--rw ssh-keys
| | +--rw ssh-key* [name]
| | | +--rw name              string
| | | +--rw ssh-key-value     string
| | +--ro test-agent?        -> /accounts/account/test-agents/test-agent/name

```

Test Agents

☒ ☐
☒ ☐

 Tags ☐

You currently don't have any Test Agents registered. [Download and install](#) a new Test Agent.

Test Agents shared with me

No Test Agents are currently shared with you.

```

<accounts xmlns="http://ncc.netrounds.com">
  <account>
    <name>demo</name>
    <test-agents>
      <test-agent>
        <name>vtal</name>
        <description/>
        <ntp>
          <interface>eth0</interface>
          <server>time.google.com</server>
        </ntp>
        <management>
          <interface>eth0</interface>
        </management>
        <interfaces>
          <interface>
            <name>eth0</name>
            <description/>
            <interface-type>physical</interface-type>
            <mac>fa:16:3e:ec:b3:a8</mac>
            <mtu>1400</mtu>
            <speed>auto</speed>
            <interface-config>
              <ipv4-address>
                <dhcp/>
              </ipv4-address>
              <ipv6-address></ipv6-address>
            </interface-config>
          </interface>
        </interfaces>
        <tags>
          <tag>
            <name>Tag 1</name>
          </tag>
        </tags>
      </test-agent>
    </test-agents>
  </account>
</accounts>
</config>
""" .format(
account=args.netrounds_account,
testagent=args.test_agent_name

```

```
# Create Test Agent Application in Control Center
xml = """<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <accounts xmlns="http://ncc.netrounds.com">
    <account>
      <name>demo</name>
      <test-agents>
        <test-agent>
          <name>vta2</name>
          <description/>
          <tags>
            <tag>
              <name>Tag 2</name>
            </tag>
          </tags>
        </test-agent>
      </test-agents>
    </account>
  </accounts>
</config>"""
```

Test Agents

[Download](#)

Interface info

License info

i Test Agents registered within the last 24 hours (only last 5 shown):
vta1

Test Agents

☒ ☐
[Clear](#)
 Tags ☐
[All](#)
[Online](#)
[Offline](#)
[In use](#)
[Free](#)

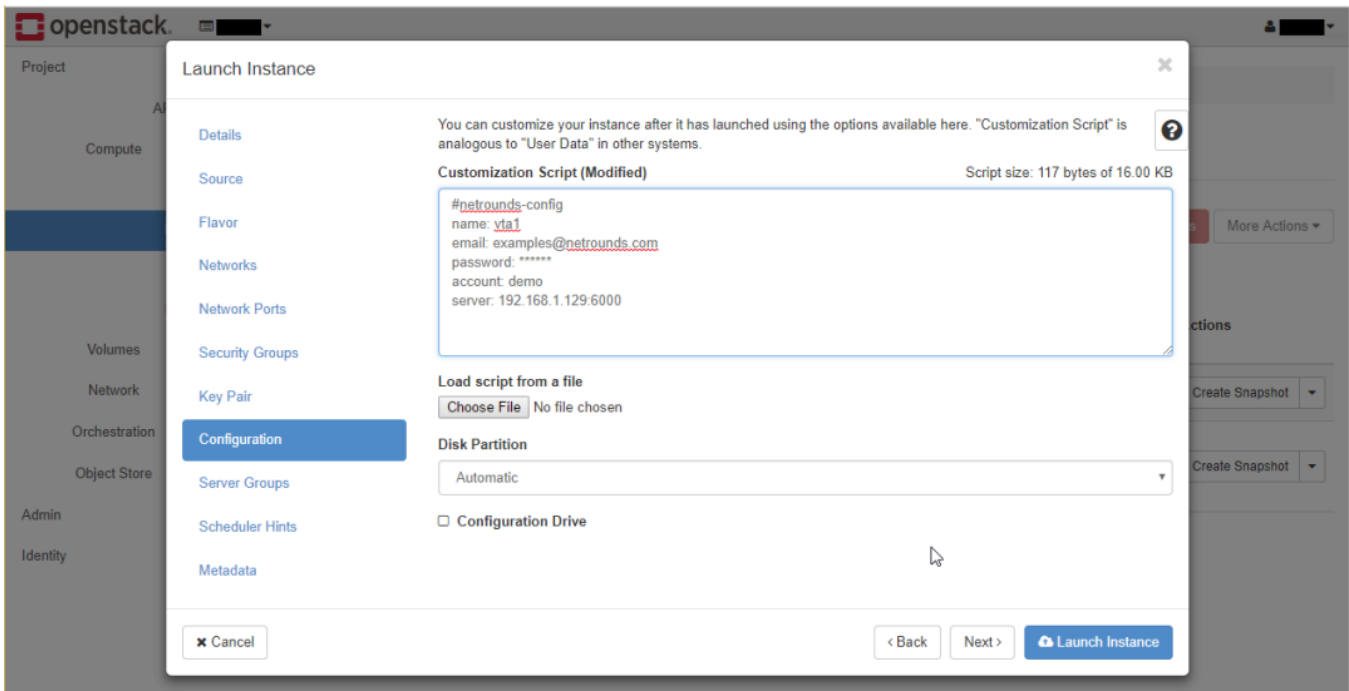
Name	Description	Management IPv4	Management IPv6	Public IP	Applications i	Share
☐  vta1	-	-	-			

```
#cloud-config
```

```
netrounds_test_agent:
```

```
  name: MyvTA                                # Name of vTA to appear in Control Center inventory
  email: john.doe@example.com                 # Email you use when logging in to the system
  password: secret                           # Login password
  account: theaccount                         # Account name
  server: <login-server>:<port>               # Control Center host and port (default == SaaS)
```

```
# Note: With an IPv6 server address the whole string
#       including port must be in double quotes
```



```
import argparse

from ncclient import manager

from ncclient.xml_ import to_ele, to_xml

# (server and account details omitted)

with manager.connect(host=args.host, port=args.port, username=args.username,
password=args.password, hostkey_verify=False) as m:

    # Get config subtree

    # Get Test Agents from account

    filter = """<accounts xmlns="http://ncc.netrounds.com">
```

```

        <account>
            <name>{account}</name>
            <test-agents></test-agents>
        </account>
    </accounts>""".format(account=args.netrounds_account)
ele = to_ele(m.get_config(source='running', filter=('subtree', xml)).data_xml)
print to_xml(ele, encoding='UTF-8', pretty_print=True)

```

```

<?xml version="1.0" ?>
<data xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <accounts xmlns="http://ncc.netrounds.com">
    <account>
      <name xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">demo</name>
      <test-agents>
        <test-agent>
          <name>vta1</name>
          <description/>
          <ntp>
            <interface>eth0</interface>
            <server>time.google.com</server>
          </ntp>
          <management>
            <interface>eth0</interface>
          </management>
          <interfaces>
            <interface>
              <name>eth0</name>
              <description/>
              <interface-type>physical</interface-type>
              <mac>fa:16:3e:ec:b3:a8</mac>
              <mtu>1400</mtu>
              <speed>auto</speed>
              <interface-config>
                <ipv4-address>
                  <dhcp/>
                </ipv4-address>
                <ipv6-address></ipv6-address>
              </interface-config>
            </interface>

```

```

          </interfaces>
        </test-agent>
      </test-agents>
    </account>
  </accounts>
</data>

```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
password=args.password,
                    hostkey_verify=False) as m:
    # Get config subtree

    # Delete Test Agent from account
    xml = """<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
        <accounts xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" xmlns="http://
ncc.netrounds.com">
            <account>
                <name>demo</name>
                <test-agents>
                    <test-agent nc:operation="delete">
                        <name>{test_agent}</name>
                    </test-agent>
                </test-agents>
            </account>
        </accounts>
</config>""".format(test_agent="vta2")

    test_agents = m.edit_config(target='running', config=xml)

```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
password=args.password,
                    hostkey_verify=False) as m:

    data = m.create_subscription(stream_name='ncc')
    # This will put ncclient in a state where it listens for notifications,
    # triggered for instance by a Test Agent being unplugged.

    # The example code below simply reads one notification. What to do with it is left
    # to the creativity of the user.
    while True:
        notification = m.take_notification()
        print notification._raw

```

```
with manager.connect(host=args.host, port=args.port, username=args.username,  
                    password=args.password, hostkey_verify=False) as m:
```

```
# Create TWAMP reflector in account
```

```
xml = """<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">  
    <accounts xmlns="http://ncc.netrounds.com">  
        <account>  
            <name>{account}</name>  
            <twamp-reflectors>  
                <twamp-reflector>  
                    <name>{name}</name>  
                    <address>{address}</address>  
                    <port>{port}</port>  
                    <control-port>{control}</control-port>  
                </twamp-reflector>  
            </twamp-reflectors>
```

```
        </account>  
    </accounts>  
</config>""".format(account=args.netrounds_account,  
                    name='My TWAMP Reflector',  
                    address='192.168.6.113',  
                    port=21000,  
                    control=862)
```

```
monitor = m.edit_config(target='running', config=xml)
```

```
with manager.connect(host=args.host, port=args.port, username=args.username,  
                    password=args.password, hostkey_verify=False) as m:
```

```
# Create Y.1731 MEP in account
```

```
xml = """<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
    <accounts xmlns="http://ncc.netrounds.com">
```

```
        <account>
```

```
            <name>{account}</name>
```

```
            <y1731-meps>
```

```
                <y1731-mep>
```

```
                    <name>{y1731_mep}</name>
```

```
                    <mac>{mac}</mac>
```

```
                    <meg-level>{meg_level}</meg-level>
```

```
                </y1731-mep>
```

```
            </y1731-meps>
```

```
        </account>
```

```
    </accounts>
```

```
</config>""".format(account=args.netrounds_account,
```

```
                    y1731_mep='Y1731',
```

```
                    mac='08:00:27:aa:95:d1',
```

```
                    meg_level=2)
```

```
monitor = m.edit_config(target='running', config=xml)
```

```

+--rw test-agents
| +--rw test-agent* [name]
|   +--rw name          string
|   +--rw description?  string
|   +--rw tags
|   | +--rw tag* [name]
|   |   +--rw name      -> ../../../../tags/tag/name
|   +--ro state
|   | +--ro status?      test-agent-status-t
|   | +--ro version?     string
|   | +--ro uptime?      int64
|   | +--ro memory-usage? percent-t
|   | +--ro load-average
|   |   +--ro last-1-minute? decimal64
|   |   +--ro last-5-minutes? decimal64
|   |   +--ro last-15-minutes? decimal64
|   +--rw ntp
|   | +--rw interface?   string
|   | +--rw servers* [address]
|   | | +--rw address     union
|   | +--rw enable-ipv6? boolean
|   +--rw management
|   | +--rw interface?   string
|   | +--rw use-public-address? boolean
|   +--rw interfaces
|   | +--rw interface* [name]
|   |   +--rw name          string
|   |   +--rw description?  string
|   |   +--rw interface-type? interface-type-t
|   |   +--rw mac?          yang:mac-address
|   |   +--rw mtu?          uint16
|   |   +--rw speed?        speed-t
|   |   +--rw bridge

```

```

| | | +--rw members
| | |   +--rw member* [interface]
| | |     +--rw interface -> ../../../../interface/name
| | |   +--rw vlan
| | |     +--rw id          vlan-id-t
| | |     +--rw parent      -> ../../../../interface/name
| | |   +--rw wifi
| | |     +--rw ssid?       string
| | |     +--rw bssid?      yang:mac-address
| | |     +--rw country?    string
| | |     +--rw ht          boolean
| | |     +--rw ht40        boolean
| | |     +--rw vht          boolean
| | |     +--rw sgi          boolean
| | |     +--rw ldpc         boolean
| | |     +--rw ht-mcs?     string
| | |     +--rw vht-max-mcs uint8
| | |     +--rw vht-max-mimo uint8
| | |     +--rw freq-2-4    boolean
| | |     +--rw freq-5      boolean
| | |     +--rw auth
| | |       +--rw (auth-type)?
| | |         +--:(personal)
| | |           +--rw personal!
| | |             +--rw cipher      cipher-auth-t
| | |             +--rw password?  string
| | |           +--:(eap-tls)
| | |             +--rw eap-tls!
| | |               +--rw cipher      cipher-auth-t
| | |               +--rw ca-cert?    string
| | |               +--rw client-cert? string
| | |               +--rw identity?   string
| | |               +--rw private-key? string
| | |           +--:(eap-ttls)
| | |             +--rw eap-ttls!
| | |               +--rw cipher      cipher-auth-t
| | |               +--rw ca-cert?    string
| | |               +--rw identity?   string
| | |               +--rw anonymous-identity? string
| | |               +--rw password?  string
| | |           +--:(peap)
| | |             +--rw peap!
| | |               +--rw cipher      cipher-auth-t

```

```

| | | +---rw ca-cert?      string
| | | +---rw identity?     string
| | | +---rw anonymous-identity? string
| | | +---rw password?     string
| | +---rw mobile
| | | +---rw apn?          string
| | | +---rw pdp_type      string
| | | +---rw rat_band      mobile-rat-band-t
| | | +---rw rat_mode      mobile-rat-mode-t
| | +---rw interface-config
| | | +---rw ipv4-address
| | | | +---rw (address-type)?
| | | | +---:(static)
| | | | | +---rw static!
| | | | | +---rw address      tailf:ipv4-address-and-prefix-length
| | | | | +---rw routes* [network]
| | | | | | +---rw network      tailf:ipv4-address-and-prefix-length
| | | | | | +---rw gateway      inet:ipv4-address
| | | | | | +---rw dns-servers* inet:ipv4-address
| | | | | +---:(dhcp)
| | | | | +---rw dhcp!
| | | | | +---rw vendor-id?    string
| | +---rw ipv6-address
| | | +---rw (address-type)?
| | | +---:(static)
| | | | +---rw static!
| | | | | +---rw address      tailf:ipv6-address-and-prefix-length
| | | | | +---rw routes* [network]
| | | | | | +---rw network      tailf:ipv6-address-and-prefix-length
| | | | | | +---rw gateway      inet:ipv6-address
| | | | | | +---rw dns-servers* inet:ipv6-address
| | | | | +---:(dhcp)
| | | | | +---rw dhcp!
| | | | | +---rw vendor-id?    string
| | | | | +---:(slaac)
| | | | | +---rw slaac!
| | | | | +---rw dns-servers*  inet:ipv6-address
| +---rw ssh-keys
| | +---rw ssh-key* [name]
| | | +---rw name            string
| | | +---rw ssh-key-value   string
| | +---ro test-agent?      -> /accounts/account/test-agents/test-agent/name

```

Test Agents



Clear

Tags

All Online Offline In use Free

You currently don't have any Test Agents registered. [Download and install](#) a new Test Agent.

Test Agents shared with me

No Test Agents are currently shared with you.

```
import argparse

from ncclient import manager

parser = argparse.ArgumentParser(description='Test creating Test Agent')
parser.add_argument('--host', help='The hostname where ConfD is found', required=True)
parser.add_argument('--port', help='The port to connect to ConfD', required=True)
parser.add_argument('--username', help='The username to connect to ConfD', required=True)
parser.add_argument('--password', help='Password to the ConfD account', required=True)
parser.add_argument('--netrounds-account', help='The NCC account short name', required=True)
parser.add_argument('--test-agent-name', help='Name of Test Agent', required=True)
args = parser.parse_args()

with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:

    # Create Test Agent in Control Center
    xml = """<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```

<accounts xmlns="http://ncc.netrounds.com">
  <account>
    <name>demo</name>
    <test-agents>
      <test-agent>
        <name>vta1</name>
        <description/>
        <ntp>
          <interface>eth0</interface>
          <server>time.google.com</server>
        </ntp>
        <management>
          <interface>eth0</interface>
        </management>
        <interfaces>
          <interface>
            <name>eth0</name>
            <description/>
            <interface-type>physical</interface-type>
            <mac>fa:16:3e:ec:b3:a8</mac>
            <mtu>1400</mtu>
            <speed>auto</speed>
            <interface-config>
              <ipv4-address>
                <dhcp/>
              </ipv4-address>
              <ipv6-address></ipv6-address>
            </interface-config>
          </interface>
        </interfaces>
        <tags>
          <tag>
            <name>Tag 1</name>
          </tag>
        </tags>
      </test-agent>
    </test-agents>
  </account>
</accounts>
</config>
""".format(
account=args.netrounds_account,
testagent=args.test_agent_name

```

```
# Create Test Agent Application in Control Center
xml = """<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <accounts xmlns="http://ncc.netrounds.com">
        <account>
            <name>demo</name>
            <test-agents>
                <test-agent>
                    <name>vta2</name>
                    <description/>
                    <tags>
                        <tag>
                            <name>Tag 2</name>
                        </tag>
                    </tags>
                </test-agent>
            </test-agents>
        </account>
    </accounts>
</config>"""
```

Test Agents

[Download](#)

[Interface info](#)
[License info](#)

Test Agents registered within the last 24 hours (only last 5 shown):
 vta1

Test Agents

☒
☐

[Clear](#)

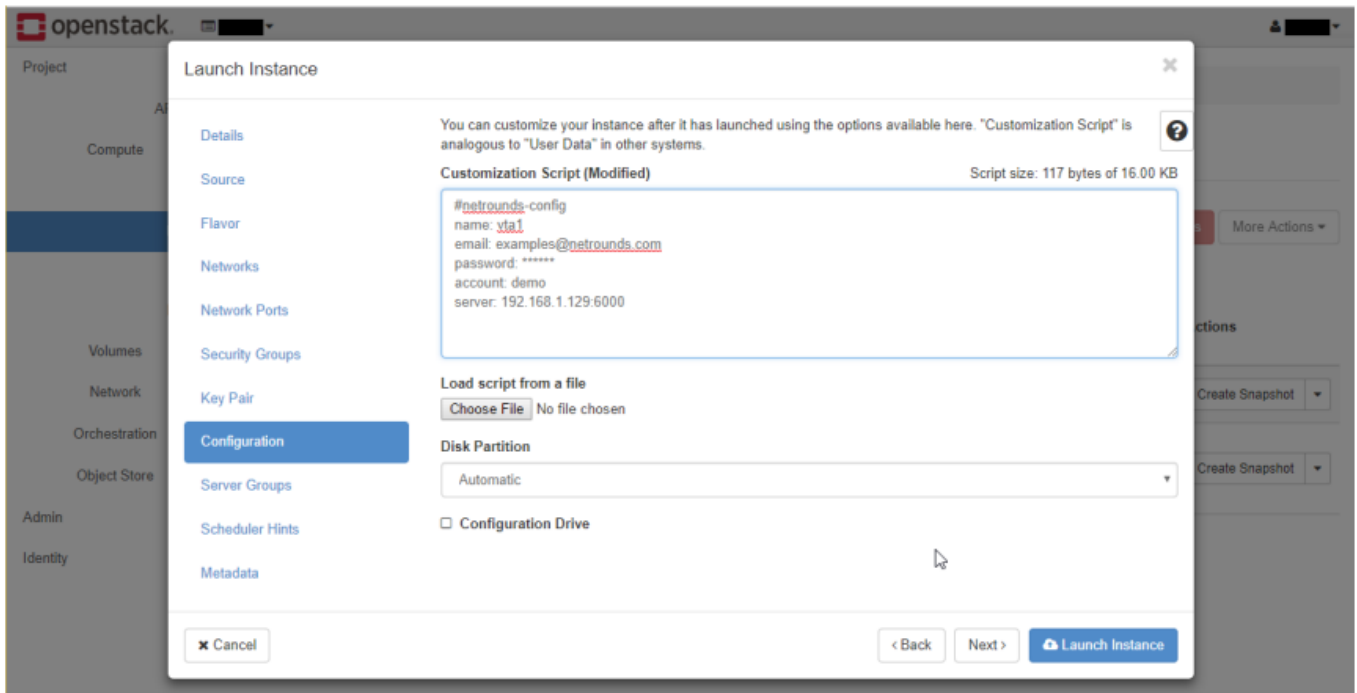
Tags ☐

[All](#)
[Online](#)
[Offline](#)
[In use](#)
[Free](#)

Name	Description	Management IPv4	Management IPv6	Public IP	Applications	Share
☐ vta1	-	-	-			

```
#cloud-config
netrounds_test_agent:
    name: MyvTA                # Name of vTA to appear in Control Center inventory
    email: john.doe@example.com # Email you use when logging in to the system
    password: secret           # Login password
    account: theaccount         # Account name
    server: <login-server>:<port> # Control Center host and port (default == SaaS)

    # Note: With an IPv6 server address the whole string
    # including port must be in double quotes
```



```
import argparse

from ncclient import manager
from ncclient.xml_ import to_ele, to_xml

# (server and account details omitted)

with manager.connect(host=args.host, port=args.port, username=args.username,
password=args.password, hostkey_verify=False) as m:

    # Get config subtree
    # Get Test Agents from account
    filter = "<accounts xmlns='http://ncc.netrounds.com'>"
```

```

        <account>
            <name>{account}</name>
            <test-agents></test-agents>
        </account>
    </accounts>""".format(account=args.netrounds_account)
    ele = to_ele(m.get_config(source='running', filter=('subtree', xml)).data_xml)
    print to_xml(ele, encoding='UTF-8', pretty_print=True)

```

Running this code gives output like that below:

```

<?xml version="1.0" ?>
<data xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <accounts xmlns="http://ncc.netrounds.com">
    <account>
      <name xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">demo</name>
      <test-agents>
        <test-agent>
          <name>vtal</name>
          <description/>
          <ntp>
            <interface>eth0</interface>
            <server>time.google.com</server>
          </ntp>
          <management>
            <interface>eth0</interface>
          </management>
          <interfaces>
            <interface>
              <name>eth0</name>
              <description/>
              <interface-type>physical</interface-type>
              <mac>fa:16:3e:ec:b3:a8</mac>
              <mtu>1400</mtu>
              <speed>auto</speed>
              <interface-config>
                <ipv4-address>
                  <dhcp/>
                </ipv4-address>
                <ipv6-address></ipv6-address>
              </interface-config>
            </interface>

```

```

          </interfaces>
        </test-agent>
      </test-agents>
    </account>
  </accounts>
</data>

```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
password=args.password,
                    hostkey_verify=False) as m:
    # Get config subtree

    # Delete Test Agent from account
    xml = """<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
        <accounts xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" xmlns="http://
ncc.netrounds.com">
            <account>
                <name>demo</name>
                <test-agents>
                    <test-agent nc:operation="delete">
                        <name>{test_agent}</name>
                    </test-agent>
                </test-agents>
            </account>
        </accounts>
    </config>""".format(test_agent="vta2")

    test_agents = m.edit_config(target='running', config=xml)

```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
password=args.password,
                    hostkey_verify=False) as m:

    data = m.create_subscription(stream_name='ncc')
    # This will put ncclient in a state where it listens for notifications,
    # triggered for instance by a Test Agent being unplugged.

    # The example code below simply reads one notification. What to do with it is left
    # to the creativity of the user.
    while True:
        notification = m.take_notification()
        print notification._raw

```

```

<nccclient.operations.notify.Notification object at 0x7f30242b6f10>
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0"><eventTime>
2017-02-03T15:09:55.939156+00:00</eventTime>
<test-agent-status-change xmlns='http://ncc.netrounds.com'>
  <account>demo</account>
  <test-agent>HW1</test-agent>
  <status>offline</status>
</test-agent-status-change>
</notification>

```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:

```

```

# Create TWAMP reflector in account

```

```

xml = """<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <accounts xmlns="http://ncc.netrounds.com">
    <account>
      <name>{account}</name>
      <twamp-reflectors>
        <twamp-reflector>
          <name>{name}</name>
          <address>{address}</address>
          <port>{port}</port>
          <control-port>{control}</control-port>
        </twamp-reflector>
      </twamp-reflectors>

```

```

    </account>
  </accounts>
</config>""".format(account=args.netrounds_account,
                    name='My TWAMP Reflector',
                    address='192.168.6.113',
                    port=21000,
                    control=862)

```

```

monitor = m.edit_config(target='running', config=xml)

```

```
with manager.connect(host=args.host, port=args.port, username=args.username,  
                    password=args.password, hostkey_verify=False) as m:
```

```
# Create Y.1731 MEP in account
```

```
xml = """<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
    <accounts xmlns="http://ncc.netrounds.com">
```

```
        <account>
```

```
            <name>{account}</name>
```

```
            <y1731-meps>
```

```
                <y1731-mep>
```

```
                    <name>{y1731_mep}</name>
```

```
                    <mac>{mac}</mac>
```

```
                    <meg-level>{meg_level}</meg-level>
```

```
                </y1731-mep>
```

```
            </y1731-meps>
```

```
        </account>
```

```
    </accounts>
```

```
</config>""".format(account=args.netrounds_account,
```

```
                    y1731_mep='Y1731',
```

```
                    mac='08:00:27:aa:95:d1',
```

```
                    meg_level=2)
```

```
monitor = m.edit_config(target='running', config=xml)
```

```
with manager.connect(host=args.host, port=args.port, username=args.username,  
                    password=args.password, hostkey_verify=False) as m:
```

```
# Create IPTV channel in account
```

```
xml = """<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">  
    <accounts xmlns="http://ncc.netrounds.com">  
        <account>  
            <name>{account}</name>  
            <iptv-channels>  
                <iptv-channel>  
                    <name>{iptv_channel_name}</name>  
                    <ip>{ip}</ip>  
                    <source>{source}</source>  
                    <port>{port}</port>  
                    <pnum>{pnum}</pnum>  
                </iptv-channel>  
            </iptv-channels>  
        </account>  
    </accounts>  
</config>""".format(account=args.netrounds_account,  
                    iptv_channel_name='My TV Channel',  
                    ip='226.226.226.0',  
                    source='226.226.226.0',  
                    port=123,  
                    pnum=123)
```

```
monitor = m.edit_config(target='running', config=xml)
```

```
with manager.connect(host=args.host, port=args.port, username=args.username,  
                    password=args.password, hostkey_verify=False) as m:
```

```
# Create Ping host in account
```

```
xml = """<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">  
    <accounts xmlns="http://ncc.netrounds.com">
```

```

        <account>
            <name>{account}</name>
            <ping-hosts>
                <ping-host>
                    <name>{name}</name>
                    <host>{host}</host>
                    <gps-lat>{gps_lat}</gps-lat>
                    <gps-long>{gps_long}</gps-long>
                </ping-host>
            </ping-hosts>
        </account>
    </accounts>
</config>""".format(account=args.netrounds_account,
                      name='My Ping Host',
                      host='192.168.6.113',
                      gps_lat=22.11,
                      gps_long=-40.40)

```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:

```

```

# Create SIP account in product account

```

```

xml = """<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <accounts xmlns="http://ncc.netrounds.com">
        <account>
            <name>{account}</name>
            <sip-accounts>
                <sip-account>
                    <sip-address>{sip_address}</sip-address>
                    <password>{password}</password>
                    <registrar>{registrar}</registrar>
                    <proxy>{proxy}</proxy>
                    <user-auth>{user_auth}</user-auth>
                    <uri-rewrite>{uri_rewrite}</uri-rewrite>
                </sip-account>
            </sip-accounts>
        </account>
    </accounts>
</config>

```

```

        </accounts>
    </config>""".format(account=args.netrounds_account,
                          sip_address='user_name@example.com',
                          password='my_password',
                          registrar='8.8.8.8',
                          proxy='example.com',
                          user_auth='my_auth_id',
                          uri_rewrite='abc')

    monitor = m.edit_config(target='running', config=xml)

```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify = False) as m:

    # Get inventory items from account
    xml = """<accounts xmlns="http://ncc.netrounds.com">
        <account>
            <name>{account}</name>
            <twamp-reflectors></twamp-reflectors>
            <y1731-meps></y1731-meps>
            <iptv-channels></iptv-channels>
            <sip-accounts></sip-accounts>
        </account>
    </accounts>""".format(account=args.netrounds_account)

    # Convert to ElementTree object
    ele = to_ele(m.get_config(source='running', filter=('subtree', xml)).data_xml)

    # Convert back to XML string and print
    print to_xml(ele, encoding='UTF-8', pretty_print=True)

```

```

<?xml version="1.0" ?>
<data xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <accounts xmlns="http://ncc.netrounds.com">
    <account>
      <name xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">demo</name>
      <twamp-reflectors>
        <twamp-reflector>
          <name>My TWAMP Reflector</name>
          <address>192.168.6.113</address>
          <port>21000</port>
          <control-port>862</control-port>
        </twamp-reflector>
      </twamp-reflectors>
      <y1731-meps>
        <y1731-mep>
          <name>Y1731</name>
          <mac>08:00:27:aa:95:d1</mac>
          <meg-level>2</meg-level>
        </y1731-mep>
      </y1731-meps>
      <iptv-channels>
        <iptv-channel>
          <name>My TV Channel</name>
          <ip>226.226.226.0</ip>
          <source>226.226.226.0</source>
          <port>123</port>
          <pnum>123</pnum>
        </iptv-channel>
      </iptv-channels>
      <sip-accounts>
        <sip-account>
          <sip-address>user_name@example.com</sip-address>
          <password>my_password</password>
          <registrar>8.8.8.8</registrar>
          <proxy>example.com</proxy>
          <user-auth>my_auth_id</user-auth>
          <uri-rewrite>abc</uri-rewrite>
        </sip-account>
      </sip-accounts>
    </account>
  </accounts>
</data>

```

```

</account>

```

```

</accounts>

```

```

</data>

```

```
with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:
```

```
# Create email list in account
```

```
xml = """<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <accounts xmlns="http://ncc.netrounds.com">
        <account>
            <name>{account}</name>
            <alarm-email-lists>
                <alarm-email-list>
                    <name>{name}</name>
```

```
                <addresses>{addresses1}</addresses>
                <addresses>{addresses2}</addresses>
            </alarm-email-list>
        </alarm-email-lists>
    </account>
</accounts>
</config>""".format(account=args.netrounds_account,
                    name='email list',
                    addresses1='thranduil@example.com',
                    addresses2='nimrodel@example.com')
```

```
print m.edit_config(target='running', config=xml)
```

```
with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:
```

```
# Get email lists in account
```

```
xml = """<accounts xmlns="http://ncc.netrounds.com">
    <account>
        <name>{account}</name>
        <alarm-email-lists></alarm-email-lists>
    </account>
</accounts>""".format(account=args.netrounds_account)
```

```
# Convert to ElementTree object
```

```
ele = to_ele(m.get_config(source='running', filter=('subtree', xml)).data_xml)
```

```
# Convert back to XML string and print
```

```
print to_xml(ele, encoding='UTF-8', pretty_print=True)
```

```
with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:
```

```

# Create SNMP manager in account (version 2c)
xml = """<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <accounts xmlns="http://ncc.netrounds.com">
        <account>
            <name>{account}</name>
            <snmp-managers>
                <snmp-manager>
                    <name>{name}</name>
                    <ip>{ip}</ip>
                    <version>{version}</version>
                    <community>{community}</community>
                </snmp-manager>
            </snmp-managers>
        </account>
    </accounts>
</config>""".format(account=args.netrounds_account,
                      version='2c',
                      ip='8.8.8.8',
                      name='snmp manager',
                      community='my community string')

```

Note: The user can also create a version 3 SNMP manager with the following parameters:

```

# <engine-id>{engine_id}</engine-id>
# <user-name>{user_name}</user-name>
# <security>{security}</security>
# <auth-password>{auth_password}</auth-password>
# <priv-password>{priv_password}</priv-password>

```

```

print m.edit_config(target='running', config=xml)

```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:

```

```

# Get SNMP managers in account

```

```

xml = """<accounts xmlns="http://ncc.netrounds.com">
    <account>
        <name>{account}</name>

```

```

        <snmp-managers></snmp-managers>
    </account>
</accounts>""".format(account=args.netrounds_account)

```

```

# Convert to ElementTree object
ele = to_ele(m.get_config(source='running', filter=('subtree', xml)).data_xml)

# Convert back to XML string and print
print to_xml(ele, encoding='UTF-8', pretty_print=True)

```

Note: This example requires existing an SNMP manager and alarm email list in order to work

with manager.connect(host=args.host, port=args.port, username=args.username, password=args.password, hostkey_verify=False) as m:

```

# Create alarm template in account
xml = """<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <accounts xmlns="http://ncc.netrounds.com">
        <account>
            <name>{account}</name>
            <alarm-templates>
                <alarm-template>
                    <name>{name}</name>
                    <email>{email}</email>
                    <snmp>{snmp}</snmp>
                    <action>{action}</action>
                    <window-size>{window_size}</window-size>
                    <interval>{interval}</interval>
                    <send-only-once>{send_only_once}</send-only-once>
                    <snmp-trap-per-stream>{snmp_trap_per_stream}</snmp-trap-per-
stream>

                    <thr-es-critical>{thr_es_critical}</thr-es-critical>
                    <thr-es-critical-clear>{thr_es_critical_clear}</thr-es-critical-
clear>

                    <thr-es-major>{thr_es_major}</thr-es-major>

```

```

        <thr-es-major-clear>{thr_es_major_clear}</thr-es-major-clear>
        <thr-es-minor>{thr_es_minor}</thr-es-minor>
        <thr-es-minor-clear>{thr_es_minor_clear}</thr-es-minor-clear>
        <thr-es-warning>{thr_es_warning}</thr-es-warning>
        <thr-es-warning-clear>{thr_es_warning_clear}</thr-es-warning-
clear>

        </alarm-template>
    </alarm-templates>
</account>
</accounts>
</config>"".format(account=args.netrounds_account,
                    name='template',
                    email='email list',
                    snmp='snmp manager',
                    action='some action string',
                    window_size=60,
                    interval=300,
                    send_only_once='false',
                    snmp_trap_per_stream='false',
                    thr_es_critical=9,
                    thr_es_critical_clear=8,
                    thr_es_major=7,
                    thr_es_major_clear=6,
                    thr_es_minor=5,
                    thr_es_minor_clear=4,
                    thr_es_warning=3,
                    thr_es_warning_clear=2)

# Note: The user can also provide additional parameters:

# <no-data-severity>{no_data_severity}</no-data-severity>
# <no-data-timeout>{no_data_timeout}</no-data-timeout>

print m.edit_config(target='running', config=xml)

```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:

```

```

# Get alarm templates in account

```

```

xml = "<accounts xmlns='http://ncc.netrounds.com'>
    <account>
        <name>{account}</name>
        <alarm-templates></alarm-templates>
    </account>
</accounts>"".format(account=args.netrounds_account)

# Convert to ElementTree object
ele = to_ele(m.get_config(source='running', filter=('subtree', xml)).data_xml)

# Convert back to XML string and print
print to_xml(ele, encoding='UTF-8', pretty_print=True)

```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:

    xml = """
    <config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <accounts xmlns="http://ncc.netrounds.com">
        <account>
          <name>{account}</name>
          <test-agents>
            <test-agent>
              <name>{test_agent}</name>
              <ssh-keys>
                <ssh-key>
                  <name>{key_name}</name>
                  <ssh-key-value>{key_value}</ssh-key-value>
                </ssh-key>
              </ssh-key>
            </test-agent>
          </test-agents>
        </account>
      </accounts>
    </config>""".format(account=args.netrounds_account,
                        key_name=args.key_name,
                        key_value=args.key_value,
                        test_agent=args.test_agent)

    monitor = m.edit_config(target='running', config=xml, default_operation='merge')

    print "ok" if monitor.ok else monitor.errors

```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:

    xml = """
    <config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <accounts xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" xmlns="http://
ncc.netrounds.com">
        <account>
          <name>{account}</name>
          <test-agents>
            <test-agent>
              <name>{test_agent}</name>
              <ssh-keys>
                <ssh-key nc:operation="delete">
                  <name>{key_name}</name>
                </ssh-key>
              </ssh-key>
            </test-agent>
          </test-agents>
        </account>
      </accounts>
    </config>""".format(account=args.netrounds_account,
                        key_name=args.key_name,
                        test_agent=args.test_agent)

    monitor = m.edit_config(target='running', config=xml, default_operation='merge')

    print "ok" if monitor.ok else monitor.errors

```

```

+---x start-test
| +---w input
| | +---w name          string
| | +---w account      -> /accounts/account/name
| | +---w template     string
| | +---w parameters
| |   +---w parameter* [key]
| |     +---w key              string
| |     +---w (value-type)
| |       +---:(integer)
| |         | +---w integer?          int64

```

```

| |      +--:(float)
| |      | +---w float?          decimal64
| |      +--:(string)
| |      | +---w string?         string
| |      +--:(test-agent-interfaces)
| |      | +---w test-agent-interfaces
| |      | +---w test-agent-interface* [account test-agent interface]
| |      | +---w account         -> ../../../../../../account
| |      | +---w test-agent      -> deref ../../../../../../account)/../test-agents/test-
agent/name
| |      | +---w interface       -> deref ../test-agent)/../interfaces/interface/name
| |      | +---w ip-version?     inet:ip-version
| |      +--:(twamp-reflectors)
| |      | +---w twamp-reflectors
| |      | +---w twamp-reflector* [name]
| |      | +---w name           -> deref ../../../../../../account)/../twamp-reflectors/twamp-
reflector/name
| |      +--:(y1731-meps)
| |      | +---w y1731-meps
| |      | +---w y1731-mep* [name]
| |      | +---w name           -> deref ../../../../../../account)/../y1731-meps/y1731-mep/
name
| |      +--:(iptv-channels)
| |      | +---w iptv-channels
| |      | +---w iptv-channel* [name]
| |      | +---w name           -> deref ../../../../../../account)/../iptv-channels/iptv-
channel/name
| |      +--:(sip-accounts)
| |      | +---w sip-accounts
| |      | +---w sip-account* [account test-agent interface sip-address]
| |      | +---w account         -> ../../../../../../account
| |      | +---w test-agent      -> deref ../../../../../../account)/../test-agents/
test-agent/name
| |      | +---w interface       -> deref ../test-agent)/../interfaces/interface/name
| |      | +---w sip-address     -> deref ../../../../../../account)/../sip-accounts/
sip-account/sip-address
| +--ro output
|   +--ro result      test-status-t
|   +--ro id?         uint32
|   +--ro error-text? string

```

Tests

[Show all](#)

You are currently not running any tests. [Create new test](#) or use [QuickStart](#).

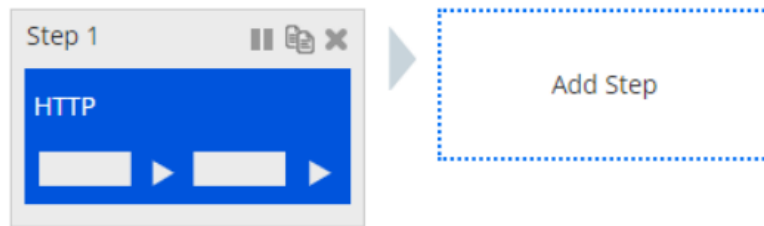
Started

Share

New test sequence

Name:

Description:



▼ Step 1

HTTP

Step 1



▼ General

Clients ⓘ

Select interfaces

URL ⓘ

```

with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:

    # Run RPC "start-test"

    # A template named "HTTP_test" and having two inputs is assumed to exist
    xml = """<start-test xmlns="http://ncc.netrounds.com">
        <name>Netconf Initiated HTTP test</name>
        <account>{account}</account>
        <template>HTTP_test</template>
        <parameters>
            <parameter>
                <key>clients</key>
                <test-agent-interfaces>
                    <test-agent-interface>
                        <account>{account}</account>
                        <test-agent>TA1</test-agent>
                        <interface>eth0</interface>
                    </test-agent-interface>
                </test-agent-interfaces>
            </parameter>
            <parameter>
                <key>url</key>
                <string>http://www.google.se</string>
            </parameter>
        </parameters>
    </start-test>""".format(account=args.netrounds_account)

    # Convert to ElementTree element
    elem = to_ele(xml)

    print m.dispatch(elem)

```

NETCONF-initiated HTTP test

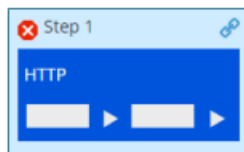
[Click here to add a description]

Rerun

Delete

Report

Export



HTTP

Client

ES history

Response time average
(ms)

First byte received
(ms)

Rate
(Mbit/s)

VTA1:eth0 (IPv4)

209

209

0.04824

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-
id="urn:uuid:f08efc54-2086-4d09-af71-5b65fcecabe1"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"><id xmlns='http://ncc.netrounds.com'>3</id>
</rpc-reply>
```

```
# Get results from test in account
xml = """<accounts xmlns="http://ncc.netrounds.com">
    <account>
        <name>{account}</name>
        <tests>
            <test>
                <id>{id}</id>
            </test>
        </tests>
    </account>
</accounts>""".format(account=args.netrounds_account,
                        id=3)

# Convert to ElementTree object
ele = to_ele(m.get(filter=('subtree', xml)).data_xml)

# Convert back to XML string and print
print to_xml(ele, encoding='UTF-8', pretty_print=True)
```

```

<?xml version="1.0" ?>
    <data xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
        <accounts xmlns="http://ncc.netrounds.com">
            <account>
                <name xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">demo</name>
                <tests>
                    <test>
                        <id xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">3</id>
                        <name>HTTP</name>
                        <status>passed</status>
                        <end-time>2017-01-26T22:23:53+00:00</end-time>
                        <report-url>https://localhost//demo/testing/3/</report-url>
                        <steps>
                            <step>
                                <id>4</id>
                                <name>Step 1</name>
                                <start-time>2017-01-26T22:22:51+00:00</start-time>
                                <end-time>2017-01-26T22:23:51+00:00</end-time>
                                <status>passed</status>
                                <status-message/>
                            </step>
                        </steps>
                    </test>
                </tests>
            </account>
        </accounts>
    </data>

```

```
with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:

    # Export selected templates from Control Center
    # In order to export all templates, remove the <templates></templates> node
    export_xml = ""
    <export-test-templates xmlns="http://ncc.netrounds.com">
        <account>{account}</account>
        <templates>
            <template>
                <name>template_udp</name>
            </template>
            <template>
                <name>template_tcp</name>
            </template>
        </templates>
    </export-test-templates>"".format(account=args.netrounds_account)

    elem = to_ele(export_xml)
    response = m.dispatch(elem)
```

```
with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:
```

```
    json_string = """
        {
            "templates":[
                {
                    "input":[
                        {
                            "require_bridge":false,
                            # ...
                            # Insert rest of JSON string here
                            # ...
                        }
                    ],
                    "version": "2.29",
                    "export_date": "2019-05-29T12:39:31.172594Z"
                }
            ]
        }
    """
```

```
# Import templates back into Control Center
```

```
import_xml = """
    <import-test-templates xmlns="http://ncc.netrounds.com">
        <account>{account}</account>
        <overwrite>false</overwrite>
        <data>{data}</data>
    </import-test-templates>""".format(
        account=args.netrounds_account,
        data=json_string)
```

```
elem = to_ele(import_xml)
response = m.dispatch(elem)
```

```
print response
```

▼ General

Clients ⓘ

Select interfaces

URL ⓘ

```
with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:
```

```
# Apply configuration for a monitor, creating a monitor in Control Center
```

```
xml = """<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <accounts xmlns="http://ncc.netrounds.com">
    <account>
      <name>{account}</name>
      <monitors>
        <monitor>
          <name>{name}</name>
          <template>{template}</template>
          <parameters>
            <parameter>
              <key>url</key>
              <string>{url}</string>
            </parameter>
            <parameter>
              <key>clients</key>
              <test-agent-interfaces>
                <test-agent-interface>
                  <account>{account}</account>
                  <test-agent>{testagent}</test-agent>
                  <interface>{interface}</interface>
                  <ip-version>ipv4</ip-version>
                </test-agent-interface>
              </test-agent-interfaces>
            </parameter>
          </parameters>
        </monitor>
      </monitors>
    </account>
  </accounts>
</config>""".format(account=args.netrounds_account,
                    name='Web sites monitor',
                    template='HTTP',
                    testagent='vta1',
                    interface='eth0',
```

```
url='http://example.com',server='vTA1',  
)
```

```
monitor = m.edit_config(target='running', config=xml)
```

Run this script using ncclient. Provided that all is correct, the monitor will be initiated and started:

Web sites monitor
[\[Click here to add a description\]](#)

[Edit](#) [Clone](#) [Report](#) [Export](#) [Delete](#)

Using template "HTTP monitor" 

● HTTP



Client	ES history	Response time average (ms)	First byte received (ms)	Rate (Mbit/s)
VTA1:eth0 (IPv4)		198	198	0.051

Note: This example requires an existing SNMP manager, alarm email list and monitor template in order to work

with `manager.connect(host=args.host, port=args.port, username=args.username, password=args.password, hostkey_verify=False)` as `m`:

```
# Apply configuration for a monitor, creating a monitor in Control Center  
xml = """<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">  
  <accounts xmlns="http://ncc.netrounds.com">  
    <account>  
      <name>{account}</name>  
      <monitors>
```

```

        <monitor>
            <name>{name}</name>
            <template>{template}</template>
            <alarm-configs>
                <alarm-config>
                    <identifier>{id}</identifier>
                    <template>{alarm_template}</template>
                    <thr-es-critical>{thr_es_critical}</thr-es-critical>
                </alarm-config>
            </alarm-configs>
        </monitor>
    </monitors>
</account>
</accounts>
</config>""".format(account=args.netrounds_account,
                      name='HTTP monitor',
                      template='name_of_monitor_template',
                      id='name_of_alarm_config',    # Enter a name for the alarm
configuration here

                      alarm_template='name_of_alarm_template',
                      thr_es_critical=20)          # The template value will be
overridden by this one

    print m.edit_config(target='running', config=xml)

```

Note: This example requires an existing SNMP manager, alarm email list and monitor template in order to work

with `manager.connect(host=args.host, port=args.port, username=args.username, password=args.password, hostkey_verify=False)` as `m`:

Apply configuration for a monitor, creating a monitor in Control Center

```

xml = """<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <accounts xmlns="http://ncc.netrounds.com">
        <account>
            <name>{account}</name>

```

```

<monitors>
  <monitor>
    <name>{name}</name>
    <template>{template}</template>
    <parameters>
      <parameter>
        <key>url</key>
        <string>{url}</string>
      </parameter>
      <parameter>
        <key>clients</key>
        <test-agent-interfaces>
          <test-agent-interface>
            <account>{account}</account>
            <test-agent>{testagent}</test-agent>
            <interface>{interface}</interface>
            <ip-version>ipv4</ip-version>
          </test-agent-interface>
        </test-agent-interfaces>
      </parameter>
    </parameters>
    <alarm-configs>
      <alarm-config>
        <identifier>{id}</identifier>
        <email>{email}</email>
        <snmp>{snmp}</snmp>
        <action>{action}</action>
        <window-size>{window_size}</window-size>
        <interval>{interval}</interval>
        <send-only-once>{send_only_once}</send-only-once>
        <snmp-trap-per-stream>{snmp_trap_per_stream}</snmp-trap-
per-stream>

        <thr-es-critical>{thr_es_critical}</thr-es-critical>
        <thr-es-critical-clear>{thr_es_critical_clear}</thr-es-
critical-clear>

        <thr-es-major>{thr_es_major}</thr-es-major>
        <thr-es-major-clear>{thr_es_major_clear}</thr-es-major-
clear>

        <thr-es-minor>{thr_es_minor}</thr-es-minor>
        <thr-es-minor-clear>{thr_es_minor_clear}</thr-es-minor-
clear>

        <thr-es-warning>{thr_es_warning}</thr-es-warning>
        <thr-es-warning-clear>{thr_es_warning_clear}</thr-es-

```

```

warning-clear>
        </alarm-config>
    </alarm-configs>
</monitor>
</monitors>
</account>
</accounts>
</config>""".format(account=args.netrounds_account,
                      name='HTTP monitor',
                      template='name_of_monitor_template',
                      testagent='vta1',
                      interface='eth0',
                      url='http://example.com',
                      id='alarm_config',
                      email='email_list',
                      snmp='snmp_manager',
                      action='some_action_string',
                      window_size=60,
                      interval=300,
                      send_only_once='false',
                      snmp_trap_per_stream='false',
                      thr_es_critical=9,
                      thr_es_critical_clear=8,
                      thr_es_major=7,
                      thr_es_major_clear=6,
                      thr_es_minor=5,
                      thr_es_minor_clear=4,
                      thr_es_warning=3,
                      thr_es_warning_clear=2)

print m.edit_config(target='running', config=xml)

```

```

# Get configuration for running monitors in account
xml = """<accounts xmlns="http://ncc.netrounds.com">
    <account>
        <name>{account}</name>
        <monitors>
            <monitor>
            </monitor>
        </monitors>
    </account>
</accounts>""".format(account=args.netrounds_account)

monitor = m.get_config(source='running', filter=('subtree', xml)).data_xml

# Convert to ElementTree object
ele = to_ele(monitor)

# Convert back to XML string and print

print to_xml(ele, encoding='UTF-8', pretty_print=True)

```

```

<?xml version="1.0" ?>
  <data xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
    <accounts xmlns="http://ncc.netrounds.com">
      <account>
        <name xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">demo</name>
        <monitors>
          <monitor>
            <name>Network Quality</name>
            <template>UDP</template>
            <parameters>
              <parameter>
                <key>clients</key>
                <test-agent-interfaces>
                  <test-agent-interface>
                    <account>demo</account>
                    <test-agent>vTA1</test-agent>
                    <interface>eth1</interface>
                    <ip-version>ipv4</ip-version>
                  </test-agent-interface>

```

```

                  </test-agent-interfaces>
                </parameter>
              <parameter>
                <key>server</key>
                <test-agent-interfaces>
                  <test-agent-interface>
                    <account>demo</account>
                    <test-agent>vTA2</test-agent>
                    <interface>eth1</interface>
                    <ip-version>ipv4</ip-version>
                  </test-agent-interface>
                </test-agent-interfaces>
              </parameter>
            </parameters>
          </monitor>
        </monitors>
      </account>
    </accounts>
  </data>

```

```
with manager.connect(host=args.host, port=args.port, username=args.username,  
                    password=args.password, hostkey_verify=False) as m:
```

```
# Get status of monitor named "Network Quality" in account
```

```
xml = "<<<<<accounts xmlns='http://ncc.netrounds.com'>
```

```
    <account>
```

```
        <name>{account}</name>
```

```
        <monitors>
```

```
            <monitor>
```

```
                <name>Network Quality</name>
```

```
                <status></status>
```

```
            </monitor>
```

```
        </monitors>
```

```
    </account>
```

```
</accounts>>>>>"".format(account=args.netrounds_account)
```

```
monitor = m.get(filter=('subtree', xml)).data_xml
```

```
# Convert to ElementTree object
```

```
ele = to_ele(monitor)
```

```
# Convert back to XML string and print
```

```
print to_xml(ele, encoding='UTF-8', pretty_print=True)
```

```

<?xml version="1.0" ?>
<data xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <accounts xmlns="http://ncc.netrounds.com">
    <account>
      <name xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">demo</name>
      <monitors>
        <monitor>
          <name xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">Network Quality</
name>
          <status>
            <id>27</id>
            <last-15-minutes>
              <status>good</status>
              <status-value>100.0</status-value>
            </last-15-minutes>
            <last-hour>
              <status>good</status>
              <status-value>100.0</status-value>
            </last-hour>
            <last-24-hours>
              <status>good</status>
              <status-value>100.0</status-value>
            </last-24-hours>
          </status>
        </monitor>
      </monitors>
    </account>

```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:

```

```

# Export selected templates from Control Center
# In order to export all templates, remove the <templates></templates> node
export_xml = """
  <export-monitor-templates xmlns="http://ncc.netrounds.com">
    <account>{account}</account>
    <templates>
      <template>
        <name>template_a</name>
      </template>
      <template>

```

```

        <name>template_b</name>
    </template>
</templates>
</export-monitor-templates>""".format(account=args.netrounds_account)

elem = to_ele(export_xml)
response = m.dispatch(elem)

# Get json config from response
root = ET.fromstring(response._raw)
json_config = root[0].text

print json_config

```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:

```

```

    json_string = """
    {
        "templates":[
            {
                "input":[
                    {
                        "require_bridge":false,
                        # ...
                        # Insert rest of JSON string here
                        # ...
                    }
                ],
                "version":"2.29",
                "export_date":"2019-05-29T12:34:47.834430Z"
            }
        ]
    }
    """

```

```

# Import templates back to Control Center
import_xml = """
    <import-monitor-templates xmlns="http://ncc.netrounds.com">

```

```

        <account>{account}</account>
        <overwrite>false</overwrite>
        <data>{data}</data>
    </import-monitor-templates>""".format(account=args.netrounds_account,
                                           data=json_string)

    elem = to_ele(import_xml)
    response = m.dispatch(elem)

    print response

```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:

```

```

xml = """
<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <accounts xmlns="http://ncc.netrounds.com">
    <account>
      <name>{account}</name>
      <tags>
        <tag>
          <name>{tag}</name>
          <color>#00ff00</color>
        </tag>
      </tags>
    </account>
  </accounts>
</config>""".format(account=args.netrounds_account,
                    tag=args.tag_name)

```

```

monitor = m.edit_config(target='running', config=xml)

```

```

print "ok" if monitor.ok else monitor.errors

```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:

    xml = """
    <config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <accounts xmlns="http://ncc.netrounds.com">
        <account>
          <name>{account}</name>
          <test-agents>
            <test-agent>
              <name>vta1</name>
              <tags>
                <tag>
                  <name>{tag}</name>
                </tag>
              </tags>
            </test-agent>
          </test-agents>
        </account>
      </accounts>
    </config>""".format(account=args.netrounds_account,
                        tag=args.tag_name)

    monitor = m.edit_config(target='running', config=xml, default_operation='merge')

    print "ok" if monitor.ok else monitor.errors

```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:

    xml = """
    <config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <accounts xmlns="http://ncc.netrounds.com">
        <account>
          <name>{account}</name>
          <twamp-reflectors>
            <twamp-reflector>

```

```

        <name>{name}</name>
        <tags>
            <tag>
                <name>{tag}</name>
            </tag>
        </tags>
    </twamp-reflector>
</twamp-reflectors>
</account>
</accounts>
</config>""".format(account=args.netrounds_account,
                      name='Name',
                      tag='Tag1')

monitor = m.edit_config(target='running', config=xml, default_operation='merge')

print "ok" if monitor.ok else monitor.errors

```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:

    xml = """
<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <accounts xmlns="http://ncc.netrounds.com">
        <account>
            <name>{account}</name>
            <monitors>
                <monitor>
                    <name>{name}</name>
                    <tags>
                        <tag>
                            <name>{tag}</name>
                        </tag>
                    </tags>
                </monitor>
            </monitors>
        </account>
    </accounts>
</config>""".format(account=args.netrounds_account,
                      name='HTTP monitor',

```

```
tag='Tag 1')
```

```
monitor = m.edit_config(target='running', config=xml, default_operation='merge')
```

```
print "ok" if monitor.ok else monitor.errors
```

```
with manager.connect(host=args.host, port=args.port, username=args.username,  
                    password=args.password, hostkey_verify=False) as m:
```

```
xml = """
```

```
<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
```

```
  <accounts xmlns="http://ncc.netrounds.com">
```

```
    <account>
```

```
      <name>{account}</name>
```

```
      <tags>
```

```
        <tag>
```

```
          <name>{tag}</name>
```

```
          <color>#ff0000</color>
```

```
        </tag>
```

```
      </tags>
```

```
    </account>
```

```
  </accounts>
```

```
</config>""".format(account=args.netrounds_account, tag=args.tag_name)
```

```
monitor = m.edit_config(target='running', config=xml, default_operation='merge')
```

```
print "ok" if monitor.ok else monitor.errors
```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:

    xml = """
    <config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
        <accounts xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" xmlns="http://
ncc.netrounds.com">
            <account>
                <name>{account}</name>
                <monitors>
                    <monitor>
                        <name>{name}</name>
                        <tags>
                            <tag nc:operation="delete">
                                <name>{tag}</name>
                            </tag>
                        </tags>
                    </monitor>
                </monitors>
            </account>
        </accounts>
    </config>""".format(account=args.netrounds_account,
                        name='HTTP monitor',
                        tag='Tag 1')

    monitor = m.edit_config(target='running', config=xml, default_operation='merge')

    print "ok" if monitor.ok else monitor.errors

```

```

with manager.connect(host=args.host, port=args.port, username=args.username,
                    password=args.password, hostkey_verify=False) as m:

    xml = """
    <config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
        <accounts xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" xmlns="http://
ncc.netrounds.com">
            <account>
                <name>{account}</name>
                <tags>
                    <tag nc:operation="delete">
                        <name>{tag}</name>
                    </tag>
                </tags>
            </account>
        </accounts>
    </config>""".format(account=args.netrounds_account, tag=args.tag_name)

    monitor = m.edit_config(target='running', config=xml)

    print "ok" if monitor.ok else monitor.errors

```

Each node is printed as:

```
<status> <flags> <name> <opts> <type> <if-features>
```

<status> is one of:

- + for current
- x for deprecated
- o for obsolete

<flags> is one of:

- rw for configuration data
- ro for non-configuration data
- x for rpcs and actions
- n for notifications

<name> is the name of the node

(<name>) means that the node is a choice node

:(<name>) means that the node is a case node

If the node is augmented into the tree from another module, its name is printed as <prefix>:<name>.

<opts> is one of:

- ? for an optional leaf, choice, anydata or anyxml
- ! for a presence container
- * for a leaf-list or list

```
module: netrounds-ncc
  +--rw statistics
  | +--ro users? int32
  | +--ro users-created-24h? int32
  | +--ro users-active-1h? int32
  | +--ro test-agent-appliance? int32
  | +--ro test-agent-appliance-online? int32
  | +--ro test-agent-application? int32
  | +--ro test-agent-application-online? int32
  | +--ro monitors-started? int32
  | +--ro monitor-tasks-started? int32
  | +--ro monitor-tasks-running-http? int32
  | +--ro monitor-tasks-running-ping? int32
  | +--ro monitor-tasks-running-dns? int32
  | +--ro monitor-tasks-running-iptv? int32
  | +--ro streams-active? int32
  | +--ro accounts? int32
  | +--ro accounts-created-24h? int32
  | +--ro scheduled-calls-running? int32
  | +--ro scheduled-call-latency? decimal64
  +--rw accounts
    +--rw account* [name]
      +--rw name string
      +--rw statistics
      | +--ro users? int32
      | +--ro users-created-24h? int32
      | +--ro users-active-1h? int32
      | +--ro test-agent-appliance? int32
      | +--ro test-agent-appliance-online? int32
      | +--ro test-agent-application? int32
      | +--ro test-agent-application-online? int32
      | +--ro monitors-started? int32
```

```

| +--ro monitor-tasks-started?      int32
| +--ro monitor-tasks-running-http? int32
| +--ro monitor-tasks-running-ping? int32
| +--ro monitor-tasks-running-dns?  int32
| +--ro monitor-tasks-running-iptv?  int32
| +--ro streams-active?              int32
+--rw test-agents
| +--rw test-agent* [name]
|   +--rw name          string
|   +--rw description?   string
|   +--rw tags
|     | +--rw tag* [name]
|     |   +--rw name    -> ../../../../tags/tag/name
|   +--ro state
|     | +--ro last-update?      yang:date-and-time
|     | +--ro status?           test-agent-status-t
|     | +--ro version?          string
|     | +--ro uptime?           int64
|     | +--ro cpu-usage?        percent-t
|     | +--ro memory-usage?     percent-t
|     | +--ro interface-config-error? string
|     | +--ro update-error?     string
|     | +--ro ntp
|     | | +--ro client
|     | | | +--ro time-offset?  decimal64
|     | | | +--ro sources* [address]
|     | | |   +--ro address      string
|     | | |   +--ro status?      string
|     | | |   +--ro stratum?     int64
|     | | |   +--ro poll-interval? int64
|     | | |   +--ro reach?       string
|     | | |   +--ro last-rx?     int64
|     | +--ro load-average
|     |   +--ro last-1-minute?   decimal64
|     |   +--ro last-5-minutes?  decimal64
|     |   +--ro last-15-minutes? decimal64
|   +--rw ntp
|     | +--rw interface?   string
|     | +--rw servers* [address]
|     | | +--rw address    union
|     | +--rw enable-ipv6? boolean
|   +--rw management
|     | +--rw interface?    string

```

```

| | +--rw use-public-address?  boolean
| +--rw interfaces
| | +--rw interface* [name]
| | | +--rw name                string
| | | +--rw description?        string
| | | +--rw interface-type?     interface-type-t
| | | +--rw mac?                yang:mac-address
| | | +--rw mtu?                uint16
| | | +--rw speed?              speed-t
| | | +--rw physical
| | | | +--rw auth
| | | | | +--rw (auth-type)?
| | | | | +--:(eap-tls)
| | | | | | +--rw eap-tls!
| | | | | | | +--rw ca-cert?      string
| | | | | | | +--rw client-cert? string
| | | | | | | +--rw identity     string
| | | | | | | +--rw private-key? string
| | | | | +--:(eap-ttls)
| | | | | | +--rw eap-ttls!
| | | | | | | +--rw ca-cert?      string
| | | | | | | +--rw identity     string
| | | | | | | +--rw anonymous-identity? string
| | | | | | | +--rw password     string
| | | | | +--:(peap)
| | | | | | +--rw peap!
| | | | | | | +--rw ca-cert?      string
| | | | | | | +--rw identity     string
| | | | | | | +--rw anonymous-identity? string
| | | | | | | +--rw password     string
| | | +--rw bridge
| | | | +--rw members
| | | | | +--rw member* [interface]
| | | | | | +--rw interface -> ../../../../interface/name
| | | +--rw vlan
| | | | +--rw id                vlan-id-t
| | | | +--rw parent            -> ../../../../interface/name
| | | +--rw wifi
| | | | +--rw ssid?              string
| | | | +--rw bssid?             yang:mac-address
| | | | +--rw country?           string
| | | | +--rw ht                 boolean
| | | | +--rw ht40                boolean

```

```

| | | +--rw vht          boolean
| | | +--rw sgi          boolean
| | | +--rw ldpc         boolean
| | | +--rw ht-mcs?      string
| | | +--rw vht-max-mcs  uint8
| | | +--rw vht-max-mimo uint8
| | | +--rw freq-2-4     boolean
| | | +--rw freq-5       boolean
| | | +--rw auth
| | |   +--rw (auth-type)?
| | |     +--:(personal)
| | |       | +--rw personal!
| | |       |   +--rw cipher      cipher-auth-t
| | |       |   +--rw password    string
| | |       +--:(eap-tls)
| | |         | +--rw eap-tls!
| | |         |   +--rw cipher      cipher-auth-t
| | |         |   +--rw ca-cert?    string
| | |         |   +--rw client-cert? string
| | |         |   +--rw identity    string
| | |         |   +--rw private-key? string
| | |         +--:(eap-ttls)
| | |           | +--rw eap-ttls!
| | |           |   +--rw cipher      cipher-auth-t
| | |           |   +--rw ca-cert?    string
| | |           |   +--rw identity    string
| | |           |   +--rw anonymous-identity? string
| | |           |   +--rw password    string
| | |           +--:(peap)
| | |             +--rw peap!
| | |             +--rw cipher      cipher-auth-t
| | |             +--rw ca-cert?    string
| | |             +--rw identity    string
| | |             +--rw anonymous-identity? string
| | |             +--rw password    string
| | +--rw mobile
| |   | +--rw apn?      string
| |   | +--rw pdp_type  string
| |   | +--rw rat_band  mobile-rat-band-t
| |   | +--rw rat_mode  mobile-rat-mode-t
| |   +--rw interface-config
| |     +--rw ipv4-address
| |     | +--rw (address-type)?

```

```

| | | +--:(static)
| | | | +--rw static!
| | | | +--rw address tailf:ipv4-address-and-prefix-length
| | | | +--rw routes* [network]
| | | | | +--rw network tailf:ipv4-address-and-prefix-length
| | | | | +--rw gateway inet:ipv4-address
| | | | +--rw dns-servers* inet:ipv4-address
| | | +--:(dhcp)
| | | | +--rw dhcp!
| | | | +--rw vendor-id? string
| | +--rw ipv6-address
| | +--rw (address-type)?
| | +--:(static)
| | | +--rw static!
| | | | +--rw address tailf:ipv6-address-and-prefix-length
| | | | +--rw routes* [network]
| | | | | +--rw network tailf:ipv6-address-and-prefix-length
| | | | | +--rw gateway inet:ipv6-address
| | | | +--rw dns-servers* inet:ipv6-address
| | | +--:(dhcp)
| | | | +--rw dhcp!
| | | | +--rw vendor-id? string
| | | +--:(slaac)
| | | | +--rw slaac!
| | | | +--rw dns-servers* inet:ipv6-address
| +--rw ssh-keys
| | +--rw ssh-key* [name]
| | | +--rw name string
| | | +--rw ssh-key-value string
| | | +--ro test-agent? -> /accounts/account/test-agents/test-agent/name
+--ro registered-test-agents
| +--ro test-agent* [name]
| | +--ro name string
| | +--ro description? string
| | +--ro tags
| | | +--ro tag* [name]
| | | | +--ro name -> ../../../../tags/tag/name
| | +--ro state
| | | +--ro last-update? yang:date-and-time
| | | +--ro status? test-agent-status-t
| | | +--ro version? string
| | | +--ro uptime? int64
| | | +--ro cpu-usage? percent-t

```

```

| | +--ro memory-usage?          percent-t
| | +--ro interface-config-error? string
| | +--ro update-error?          string
| | +--ro ntp
| | | +--ro client
| | | | +--ro time-offset? decimal64
| | | | +--ro sources* [address]
| | | |   +--ro address          string
| | | |   +--ro status?          string
| | | |   +--ro stratum?         int64
| | | |   +--ro poll-interval?   int64
| | | |   +--ro reach?           string
| | | |   +--ro last-rx?         int64
| | | +--ro load-average
| | |   +--ro last-1-minute?     decimal64
| | |   +--ro last-5-minutes?    decimal64
| | |   +--ro last-15-minutes?   decimal64
| +--ro ntp
| | +--ro interface?            string
| | +--ro servers* [address]
| | | +--ro address             union
| | | +--ro enable-ipv6?        boolean
| +--ro management
| | +--ro interface?            string
| | +--ro use-public-address?    boolean
| +--ro interfaces
| | +--ro interface* [name]
| | | +--ro name                 string
| | | +--ro description?         string
| | | +--ro interface-type?      interface-type-t
| | | +--ro mac?                 yang:mac-address
| | | +--ro mtu?                 uint16
| | | +--ro speed?               speed-t
| | | +--ro physical
| | | | +--ro auth
| | | | | +--ro (auth-type)?
| | | | |   +--:(eap-tls)
| | | | | | +--ro eap-tls!
| | | | | |   +--ro ca-cert?      string
| | | | | |   +--ro client-cert?  string
| | | | | |   +--ro identity       string
| | | | | |   +--ro private-key?   string
| | | | |   +--:(eap-ttls)

```

```

| | | | +--ro eap-tls!
| | | | +--ro ca-cert?      string
| | | | +--ro identity      string
| | | | +--ro anonymous-identity? string
| | | | +--ro password      string
| | | +--:(peap)
| | | +--ro peap!
| | | +--ro ca-cert?      string
| | | +--ro identity      string
| | | +--ro anonymous-identity? string
| | | +--ro password      string
| | +--ro bridge
| | | +--ro members
| | | +--ro member* [interface]
| | | +--ro interface -> ../../../../interface/name
| | +--ro vlan
| | | +--ro id      vlan-id-t
| | | +--ro parent -> ../../../../interface/name
| | +--ro wifi
| | | +--ro ssid?      string
| | | +--ro bssid?     yang:mac-address
| | | +--ro country?   string
| | | +--ro ht         boolean
| | | +--ro ht40       boolean
| | | +--ro vht        boolean
| | | +--ro sgi        boolean
| | | +--ro ldpc       boolean
| | | +--ro ht-mcs?    string
| | | +--ro vht-max-mcs uint8
| | | +--ro vht-max-mimo uint8
| | | +--ro freq-2-4   boolean
| | | +--ro freq-5     boolean
| | | +--ro auth
| | | +--ro (auth-type)?
| | | +--:(personal)
| | | | +--ro personal!
| | | | +--ro cipher      cipher-auth-t
| | | | +--ro password    string
| | | +--:(eap-tls)
| | | | +--ro eap-tls!
| | | | +--ro cipher      cipher-auth-t
| | | | +--ro ca-cert?    string
| | | | +--ro client-cert? string

```

```

| | | | +--ro identity      string
| | | | +--ro private-key?  string
| | | +---:(eap-ttls)
| | | | +--ro eap-ttls!
| | | | +--ro cipher          cipher-auth-t
| | | | +--ro ca-cert?        string
| | | | +--ro identity        string
| | | | +--ro anonymous-identity? string
| | | | +--ro password        string
| | | +---:(peap)
| | | | +--ro peap!
| | | | +--ro cipher          cipher-auth-t
| | | | +--ro ca-cert?        string
| | | | +--ro identity        string
| | | | +--ro anonymous-identity? string
| | | | +--ro password        string
| | +--ro mobile
| | | +--ro apn?      string
| | | +--ro pdp_type  string
| | | +--ro rat_band  mobile-rat-band-t
| | | +--ro rat_mode  mobile-rat-mode-t
| | +--ro interface-config
| | | +--ro ipv4-address
| | | | +--ro (address-type)?
| | | | +---:(static)
| | | | | +--ro static!
| | | | | +--ro address      tailf:ipv4-address-and-prefix-length
| | | | | +--ro routes* [network]
| | | | | | +--ro network      tailf:ipv4-address-and-prefix-length
| | | | | | +--ro gateway      inet:ipv4-address
| | | | | +--ro dns-servers*  inet:ipv4-address
| | | | +---:(dhcp)
| | | | | +--ro dhcp!
| | | | | +--ro vendor-id?  string
| | +--ro ipv6-address
| | | +--ro (address-type)?
| | | +---:(static)
| | | | +--ro static!
| | | | +--ro address      tailf:ipv6-address-and-prefix-length
| | | | +--ro routes* [network]
| | | | | +--ro network      tailf:ipv6-address-and-prefix-length
| | | | | +--ro gateway      inet:ipv6-address
| | | | +--ro dns-servers*  inet:ipv6-address

```

```

| | +---:(dhcp)
| | | +---ro dhcp!
| | | +---ro vendor-id? string
| | +---:(slaac)
| | +---ro slaac!
| | +---ro dns-servers* inet:ipv6-address
| +---ro ssh-keys
| | +---ro ssh-key* [name]
| | | +---ro name string
| | | +---ro ssh-key-value string
| | | +---ro test-agent? -> /accounts/account/test-agents/test-agent/name
+---rw tags
| +---rw tag* [name]
| | +---rw name string
| | +---rw color? string
+---ro tests
| +---ro test* [id]
| | +---ro id uint32
| | +---ro name? string
| | +---ro status? test-status-t
| | +---ro start-time? yang:date-and-time
| | +---ro end-time? yang:date-and-time
| | +---ro report-url? string
| | +---ro report-url-pdf? string
| | +---ro steps
| | | +---ro step* [id]
| | | | +---ro id uint32
| | | | +---ro name? string
| | | | +---ro start-time? yang:date-and-time
| | | | +---ro end-time? yang:date-and-time
| | | | +---ro status? test-status-t
| | | | +---ro status-message? string
| +---ro templates
| | +---ro template* [name]
| | | +---ro name string
| | | +---ro description? string
| | | +---ro tags? string
| | | +---ro parameters
| | | | +---ro parameter* [key]
| | | | | +---ro key string
| | | | | +---ro type? parameter-type-t
+---rw twamp-reflectors
| +---rw twamp-reflector* [name]

```

```

|   +-rw name          string
|   +-rw address       string
|   +-rw ipv6?         boolean
|   +-rw port          uint16
|   +-rw control-port? uint16
|   +-rw gps-lat?      gps-lat-t
|   +-rw gps-long?     gps-long-t
|   +-rw tags
|       +-rw tag* [name]
|           +-rw name  -> ../../../../tags/tag/name
+-rw ping-hosts
|   +-rw ping-host* [name]
|       +-rw name      string
|       +-rw host      string
|       +-rw ipv6?     boolean
|       +-rw gps-lat?  gps-lat-t
|       +-rw gps-long? gps-long-t
|       +-rw tags
|           +-rw tag* [name]
|               +-rw name  -> ../../../../tags/tag/name
+-rw y1731-meps
|   +-rw y1731-mep* [name]
|       +-rw name      string
|       +-rw mac       yang:mac-address
|       +-rw meg-level uint8
+-rw iptv-channels
|   +-rw iptv-channel* [name]
|       +-rw name      string
|       +-rw ip        string
|       +-rw source?   string
|       +-rw port      uint16
|       +-rw pnum?     uint16
+-rw sip-accounts
|   +-rw sip-account* [sip-address]
|       +-rw sip-address  ascii-string-t
|       +-rw registrar?  ascii-string-t
|       +-rw password     ascii-string-t
|       +-rw proxy?       ascii-string-t
|       +-rw user-auth?   ascii-string-t
|       +-rw uri-rewrite? ascii-string-t
+-rw monitors
|   +-rw monitor* [name]
|       +-rw name      string

```

```

| | +--rw description?    string
| | +--rw started?       boolean
| | +--rw template       string
| | +--rw tags
| | | +--rw tag* [name]
| | |   +--rw name      -> ../../../../tags/tag/name
| | +--rw alarm-configs
| | | +--rw alarm-config* [identifier]
| | |   +--rw identifier      string
| | |   +--rw template?      -> ../../../../alarm-templates/alarm-template/name
| | |   +--rw email?         -> ../../../../alarm-email-lists/alarm-email-list/name
| | |   +--rw snmp?          -> ../../../../snmp-managers/snmp-manager/name
| | |   +--rw thr-es-critical? uint16
| | |   +--rw thr-es-critical-clear? uint16
| | |   +--rw thr-es-major?   uint16
| | |   +--rw thr-es-major-clear? uint16
| | |   +--rw thr-es-minor?   uint16
| | |   +--rw thr-es-minor-clear? uint16
| | |   +--rw thr-es-warning? uint16
| | |   +--rw thr-es-warning-clear? uint16
| | |   +--rw no-data-severity? uint16
| | |   +--rw no-data-timeout? uint16
| | |   +--rw action?        string
| | |   +--rw window-size?   uint16
| | |   +--rw interval?      uint16
| | |   +--rw send-only-once? boolean
| | |   +--rw snmp-trap-per-stream? boolean
| | +--rw parameters
| | | +--rw parameter* [key]
| | |   +--rw key            string
| | |   +--rw (value-type)
| | |     +--:(integer)
| | |       | +--rw integer?      int64
| | |       +--:(float)
| | |         | +--rw float?      decimal64
| | |         +--:(string)
| | |           | +--rw string?    string
| | |           +--:(test-agent-interfaces)
| | |             | +--rw test-agent-interfaces
| | |             |   +--rw test-agent-interface* [account test-agent interface]
| | |             |     +--rw account      -> ../../../../name
| | |             |     +--rw test-agent   -> ../../../../test-agents/test-agent/name
| | |             |     +--rw interface   string

```

```

| | | | +---rw ip-version? inet:ip-version
| | | | +---:(twamp-reflectors)
| | | | | +---rw twamp-reflectors
| | | | | +---rw twamp-reflector* [name]
| | | | | +---rw name -> ../../../../../../twamp-reflectors/twamp-reflector/name
| | | | +---:(ping-hosts)
| | | | | +---rw ping-hosts
| | | | | +---rw ping-host* [name]
| | | | | +---rw name -> ../../../../../../ping-hosts/ping-host/name
| | | | +---:(y1731-meps)
| | | | | +---rw y1731-meps
| | | | | +---rw y1731-mep* [name]
| | | | | +---rw name -> ../../../../../../y1731-meps/y1731-mep/name
| | | | +---:(sip-accounts)
| | | | | +---rw sip-accounts
| | | | | +---rw sip-account* [account test-agent interface sip-address]
| | | | | +---rw account -> ../../../../../../name
| | | | | +---rw test-agent -> ../../../../../../test-agents/test-agent/name
| | | | | +---rw interface -> deref(..test-agent)/../interfaces/interface/name
| | | | | +---rw sip-address -> ../../../../../../sip-accounts/sip-account/sip-

```

address

```

| | | | +---:(iptv-channels)
| | | | | +---rw iptv-channels
| | | | | +---rw iptv-channel* [name]
| | | | | +---rw name -> ../../../../../../iptv-channels/iptv-channel/name
| | +---ro status
| | | +---ro id? uint32
| | | +---ro last-15-minutes
| | | | +---ro status? monitor-sla-status-t
| | | | +---ro status-value? percent-t
| | | +---ro last-hour
| | | | +---ro status? monitor-sla-status-t
| | | | +---ro status-value? percent-t
| | | +---ro last-24-hours
| | | | +---ro status? monitor-sla-status-t
| | | | +---ro status-value? percent-t
| +---ro templates
| | +---ro template* [name]
| | | +---ro name string
| | | +---ro description? string
| | | +---ro parameters
| | | | +---ro parameter* [key]
| | | | +---ro key string

```

```

|      |      +-ro type?  parameter-type-t
|      +-ro tags?      string
+--rw alarm-templates
|  +--rw alarm-template* [name]
|      +-rw name          string
|      +-rw email?        -> ../../../../alarm-email-lists/alarm-email-list/name
|      +-rw snmp?         -> ../../../../snmp-managers/snmp-manager/name
|      +-rw thr-es-critical?  uint16
|      +-rw thr-es-critical-clear?  uint16
|      +-rw thr-es-major?    uint16
|      +-rw thr-es-major-clear?  uint16
|      +-rw thr-es-minor?    uint16
|      +-rw thr-es-minor-clear?  uint16
|      +-rw thr-es-warning?  uint16
|      +-rw thr-es-warning-clear?  uint16
|      +-rw no-data-severity?  uint16
|      +-rw no-data-timeout?  uint16
|      +-rw action?          string
|      +-rw window-size?    uint16
|      +-rw interval?       uint16
|      +-rw send-only-once?  boolean
|      +-rw snmp-trap-per-stream?  boolean
+--rw alarm-email-lists
|  +--rw alarm-email-list* [name]
|      +-rw name          string
|      +-rw addresses*    string
+--rw snmp-managers
    +--rw snmp-manager* [name]
        +-rw name          string
        +-rw ip?           string
        +-rw port?         uint16
        +-rw version?      string
        +-rw community?    string
        +-rw engine-id?    string
        +-rw user-name?    string
        +-rw security?     snmp-security-t
        +-rw auth-password? string
        +-rw priv-password? string

```

rpcs:

```

+---x start-test
|  +---w input
|  |  +---w name          string

```

```

| | +---w account      -> /accounts/account/name
| | +---w template    string
| | +---w parameters
| |   +---w parameter* [key]
| |     +---w key              string
| |     +---w (value-type)
| |       +---:(integer)
| |         | +---w integer?      int64
| |         +---:(float)
| |           | +---w float?      decimal64
| |           +---:(string)
| |             | +---w string?    string
| |             +---:(test-agent-interfaces)
| |               | +---w test-agent-interfaces
| |                 | +---w test-agent-interface* [account test-agent interface]
| |                   | +---w account      -> ../../../../account
| |                   | +---w test-agent    -> deref ../../../../account)/../test-agents/test-agent/
name
| |       | +---w interface    string
| |       | +---w ip-version?  inet:ip-version
| |       +---:(twamp-reflectors)
| |       | +---w twamp-reflectors
| |       |   +---w twamp-reflector* [name]
| |       |     +---w name      -> deref ../../../../account)/../twamp-reflectors/twamp-
reflector/name
| |       +---:(ping-hosts)
| |       | +---w ping-hosts
| |       |   +---w ping-host* [name]
| |       |     +---w name      -> deref ../../../../account)/../ping-hosts/ping-host/name
| |       +---:(y1731-meps)
| |       | +---w y1731-meps
| |       |   +---w y1731-mep* [name]
| |       |     +---w name      -> deref ../../../../account)/../y1731-meps/y1731-mep/name
| |       +---:(iptv-channels)
| |       | +---w iptv-channels
| |       |   +---w iptv-channel* [name]
| |       |     +---w name      -> deref ../../../../account)/../iptv-channels/iptv-channel/
name
| |       +---:(sip-accounts)
| |       | +---w sip-accounts
| |       |   +---w sip-account* [account test-agent interface sip-address]
| |       |     +---w account      -> ../../../../account
| |       |     +---w test-agent    -> deref ../../../../account)/../test-agents/test-agent/

```

```

name
| |          +---w interface      -> deref(..test-agent)/../interfaces/interface/name
| |          +---w sip-address    -> deref(..../../../account)/../sip-accounts/sip-
account/sip-address
| +---ro output
|   +---ro result      test-status-t
|   +---ro id?         uint32
|   +---ro error-text?  string
+---x sync-from-ncc
| +---w input
| | +---w without_interface_config?  boolean
| +---ro output
|   +---ro success      boolean
|   +---ro error-text?  string
+---x reboot-test-agents
| +---w input
| | +---w test-agents
| |   +---w (all-or-list)
| |     +---:(test-agent)
| |       | +---w test-agent* [account name]
| |       |   +---w account    -> /accounts/account/name
| |       |   +---w name      -> deref(..account)/../test-agents/test-agent/name
| |       +---:(all-in-account)
| |         +---w all-in-account? -> /accounts/account/name
| +---ro output
|   +---ro test-agents-failed* [account name]
| |   +---ro account    -> /accounts/account/name
| |   +---ro name      -> /accounts/account/test-agents/test-agent/name
| |   +---ro reason?    string
|   +---ro error-text?  string
+---x update-test-agents
| +---w input
| | +---w test-agents
| |   +---w (all-or-list)
| |     +---:(test-agent)
| |       | +---w test-agent* [account name]
| |       |   +---w account    -> /accounts/account/name
| |       |   +---w name      -> deref(..account)/../test-agents/test-agent/name
| |       +---:(all-in-account)
| |         +---w all-in-account? -> /accounts/account/name
| +---ro output
|   +---ro test-agents-failed* [account name]
| |   +---ro account    -> /accounts/account/name

```

```

| | +--ro name      -> /accounts/account/test-agents/test-agent/name
| | +--ro reason?   string
| | +--ro error-text? string
+---x sync-test-agents-stats
| +---w input
| | +---w test-agents
| |   +---w (all-or-list)
| |     +--:(test-agent)
| |       | +---w test-agent* [account name]
| |         | +---w account  -> /accounts/account/name
| |         | +---w name     -> deref(..../account)/../test-agents/test-agent/name
| |         +--:(all-in-account)
| |           +---w all-in-account? -> /accounts/account/name
| +--ro output
|   +--ro test-agents-failed* [account name]
|   | +--ro account  -> /accounts/account/name
|   | +--ro name     -> /accounts/account/test-agents/test-agent/name
|   | +--ro reason?   string
|   +--ro error-text? string
+---x export-test-templates
| +---w input
| | +---w account  -> /accounts/account/name
| | +---w templates
| |   +---w template* [name]
| |     +---w name  -> deref(..../account)/../tests/templates/template/name
| +--ro output
|   +--ro data      string
|   +--ro warnings   string
|   +--ro error-text? string
+---x export-monitor-templates
| +---w input
| | +---w account  -> /accounts/account/name
| | +---w templates
| |   +---w template* [name]
| |     +---w name  -> deref(..../account)/../monitors/templates/template/name
| +--ro output
|   +--ro data      string
|   +--ro warnings   string
|   +--ro error-text? string
+---x import-test-templates
| +---w input
| | +---w account  -> /accounts/account/name
| | +---w overwrite? boolean

```

```
| | +---w data      string
| +--ro output
|   +--ro warnings  string
|   +--ro error-text? string
+---x import-monitor-templates
    +---w input
    | +---w account  -> /accounts/account/name
    | +---w overwrite? boolean
    | +---w data      string
    +--ro output
        +--ro warnings  string
        +--ro error-text? string
```

notifications:

```
+---n test-agent-status-change
| +--ro account      -> /accounts/account/name
| +--ro test-agent   -> /accounts/account/test-agents/test-agent/name
| +--ro status       test-agent-status-t
| +--ro tags?        string
+---n test-status-change
| +--ro account      -> /accounts/account/name
| +--ro id           -> /accounts/account/tests/test/id
| +--ro status       test-status-t
| +--ro finished?    boolean
+---n monitor-sla-status-change
    +--ro account      -> /accounts/account/name
    +--ro monitor      -> /accounts/account/monitors/monitor/name
    +--ro status       monitor-sla-status-t
    +--ro prev-status  monitor-sla-status-t
    +--ro url          string
    +--ro tags?        string
```

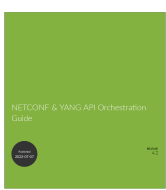
JUNIPER[®]
NETWORKS



Contents

- 1 Documents / Resources
- 1.1 References

Documents / Resources

	Juniper NETWORKS NETCONF & YANG API Software [pdf] User Guide NETCONF YANG API Software, YANG API Software, API Software, Software
---	---

References

- [Example Domain](#)
- [Google](#)
- [GitHub - mbj4668/pyang: An extensible YANG validator and converter in python](#)
- [GitHub - ncclient/ncclient: Python library for NETCONF clients](#)
- [RFC 5277 - NETCONF Event Notifications](#)
- [RFC 6241 - Network Configuration Protocol \(NETCONF\)](#)
- [User Manual](#)